



VLB REST API specifications

User documentation for retrieving metadata,
cover data and media files

Technical version: v2

Document version: 2.17

Date: 28.11.2024

Status: approved

1. Contents

1. Contents.....	2
2. Version history.....	7
3. Introduction.....	9
3.1. Target audience for this document	9
3.2. Abstract of this document	9
3.3. Further notes on this document.....	9
3.3.1. Media data users (including cover data and media data)	10
3.3.2. Cover data users	10
3.3.3. Media file users.....	10
3.4. Alternative methods for retrieving data via VLB API	10
4. VLB API technology	11
4.1. General API features.....	11
4.2. Data access functionalities	11
4.3. Interface changes relative to previous interfaces.....	12
4.3.1. Changes relative to v1 of the REST interface (HTTPS advisory)	12
4.3.2. Reference price display in API.....	12
5. Interface calling and technical integration	12
5.1. Communication with VLB API.....	12
5.1.1. URL live environment.....	13
5.1.2. URL test environment.....	13
5.2. Return formats.....	13
5.3. Field content in return formats	14
5.4. Errors	14
5.4.1. Error codes	14
5.4.2. JSON error codes	14
5.4.3. restrictions for API request (Deep paging lock-up).....	15
5.5. API login	15
5.5.1. Access token for API login	15
5.5.1.1. User-group tokens	15
5.5.1.2. Token for status-based logons.....	16
5.5.1.3. Token for statusless logons.....	16
5.5.2. Authentication.....	16

5.5.3.	Status-based login.....	17
5.5.3.1.	Login.....	17
5.5.3.2.	Logout	17
5.5.4.	Requests for media objects.....	18
5.5.5.	Notes on access token use	18
5.5.6.	Description.....	18
5.6.	Searches.....	19
5.6.1.	Quick search.....	19
5.6.2.	Boolean searches.....	19
5.6.2.1.	Search syntax and Boolean operators	21
5.6.2.2.	Important advisory concerning Boolean searches:.....	23
5.6.2.3.	Reading-rationales (Lesemotive) in the Boolean search.....	23
5.6.3.	Batch searches.....	24
5.6.4.	Request.....	24
5.6.4.1.	Return – results list.....	26
5.7.	Retrieving lists of product entries.....	28
5.7.1.	Request.....	28
5.7.2.	Return: list of product entries	28
5.8.	Product-entry retrieval.....	29
5.8.1.	Product entry returns.....	29
5.8.1.1.	JSON-long.....	29
5.8.1.2.	ONIX30 short	30
5.8.1.3.	ONIX30 ref	30
5.8.2.	Request.....	30
5.8.3.	Example of product-entry retrieval.....	31
5.9.	Accessing media files.....	31
5.9.1.	Description.....	31
5.9.2.	Request.....	35
5.9.3.	Return	35
5.9.4.	Example	35
5.10.	Cover retrieval.....	36
5.10.1.	Description.....	36
5.10.2.	Cover-Request	37

5.10.3. Return.....	37
5.11. Index searches.....	38
5.11.1. Description.....	38
5.11.2. Request	38
5.11.3. Return.....	39
5.11.4. Example.....	39
5.12. Accessing publisher data.....	39
5.12.1. Description.....	39
5.12.1. Request	40
5.12.2. Return.....	40
5.12.3. Example.....	41
5.13. Aggregators (overview).....	41
6. Examples of requests and responses.....	42
6.1. Overview of API calls	42
6.2. Requests and calls	42
6.3. API returns.....	43
6.3.1. Login-function returns.....	44
6.3.2. Responses to metadata calls.....	44
6.3.2.1. Responses to search calls.....	44
6.3.2.2. Responses to article data calls.....	45
6.3.2.3. Publisher-data call returns.....	46
6.3.3. Returns for cover and media object calls.....	46
6.4. Specimen data.....	46
7. Encoding.....	47
7.1. HTML encoding	47
7.1. URL encoding.....	47
8. Contact information	48
9. Annex.....	49
9.1. Standards and exceptions for sections and data fields.....	49
9.2. Displays of products with VAT splitting (bundled products).....	49
9.3. Specimen data.....	50
9.4. Note concerning results-lists fields	50
9.4.1. productType values	55

9.4.2. Genre codes: advisory on product-group classification	56
9.4.3. Reference price designations.....	57
9.5. Overview of the requisite ONIX code lists	57
9.5.1. Unclassified fields	57
9.5.2. taxKeyEurD / taxKeyEurA / taxKeyChf	59
9.5.3. ancillarycontents – illustrations	59
9.5.4. audiences and recommended age group	60
9.5.5. citedContents.....	60
9.5.6. Series, series and sets	61
9.5.7. contributors - Author.....	62
9.5.8. copyrights.....	63
9.5.9. Editions	64
9.5.10. extent - Extent	65
9.5.10.1. Page counts	65
9.5.10.1. DRM type	66
9.5.10.2. File size	66
9.5.10.3. Duration, track count, map scale	66
9.5.10.4. Illustrations	67
9.5.11. Product form.....	67
9.5.12. formFeatures warning	68
9.5.13. Product number identifiers	69
9.5.14. Languages/product languages	69
9.5.15. Prices.....	69
9.5.15.1.1. Taxes (included in the price block)	70
9.5.16. Prices.....	71
9.5.17. productClassifications – customs code number.....	71
9.5.18. productParts – (v1: containeditems) – product elements and ancillary elements	72
9.5.19. publisherData	72
9.5.20. Publishers.....	74
9.5.21. Related products	74
9.5.22. SalesRight	75
9.5.23. subjects – product classifications	75

9.5.24.	supportingResource media files	76
9.5.25.	textContents – ancillary texts.....	78
9.5.26.	Titles and product languages.....	79
9.5.27.	websites.....	80
9.5.28.	Wholesalers.....	80
9.5.29.	productContacts – contact address and product safety.....	80

2. Version history

Date	Version	Changes	Comments
20 June 2017	1.5		Original version
14 June 2018	1.9		Note concerning the differences between v1 and v2 See 4.3.1)
6 July 2018	2.0		finalized
18 September 2018	2.1	Note concerning the original cover (retrieving large amounts of data) promotionCampaign and marketPublishingStatus added, error messages revamped	
7 December 2018	2.3	URL corrected in section 5.6.4, ISNI, ORCID GND in 5.6.2	Status parameters in the specimen URL corrected; new parameters added to Boolean search
19 December 2018	2.4	Note concerning http and HTTPS in 4.3.1	Note concerning http/HTTPS and redirect
27 May 2019	2.5	Media file naming now based on ONIX 3	The names of certain types of media were changed and new elements of this nature were added.
19 August 2019	2.6	Code list links corrected	section 9.5.11. 9.5.24. supportingResource media files
30.07.2020	1.6	Search parameter "?" is no longer supported	Please use "*" instead of "?"
07.12.2020	2.10	API calls using Deep-Paging are blocked by the API (6.3.2.1)	if the number of pages called in the API URL is too high an error message is thrown.
10.08.2021	2.12	Reading rationales added 5.6.2.3	Reading rationales had been added to the vlb and the API using the Boolean search
22.09.2021	2.12.1	chapter 5.4.3 restrictions for API request (Deep paging lock-up) added	Error correction.compared to the german version. The deep paging chapter 5.4.3 got lost in the English translation.
29.03.2022	2.13	List of media file types has been extended	Building or game instructions and Marketing graphic were added to the table in 5.9 Accessing media files Description
31.05.2022	2.14	Text and table of media types was revised	Chapter 5.9 on the media files was revised

Date	Version	Changes	Comments
28.11.2024	2.17	Updated on the level of german specification	Chapter 5.12, 9.5.19 updated, chapter 9.5.29 added, contact data updated

3. Introduction

3.1. Target audience for this document

These specifications are intended for use by service providers who are tasked with programming access to VLP data, cover data and media files. These specifications are predicated on the assumption that the user knows how to integrate a REST interface.

3.2. Abstract of this document

- General information concerning API technology can be found in sections 3 and 4
- Interface implementation, error messages and return formats are described in sections 5.1, 5.2 and 5.3, along with the formats used to call the interface in the go-live system.
- The login interface is described in section 5.5 API login .
- Search tools and search list are described in section 5.6. These include Boolean searches and fields in search lists in the two JSON formats.
- Title-detail retrieval in various formats is described in sections 5.7 and 5.80.
- **Kindly disregard any fields that are not found in these API specifications. Such fields do not become valid until such time as they are incorporated into this document.**
- Specimen calls can be found in section 6 Examples of requests and responses
- Specimen data records in JSON format can be found via the [REST-API specimen data](#) link (Beispiele V2), where they can be accessed as zip files (6.4 note on this link)

3.3. Further notes on this document

Future versions of [API documentation](#) will be available under VLB online help.

This document contains tips and advisories on API use for the following classes of users:

1. for users of metadata, cover data and media files
2. for users who only have authorization to access cover data (cover API)
3. for users who only have authorization to access media files, including cover data

The sections of this document that are relevant for each of the various user groups are listed below.

3.3.1. Media data users (including cover data and media data)

When it comes to accessing data, metadata, cover data and/or media files, the entirety of this document applies.

3.3.2. Cover data users

Users who only access cover data will need to familiarize themselves with the material in the following sections:

- sections 1 Contents through 5.3 Field content in return formats
- section 5.5 API login
- section 5.5.1 Access token for API login
- section 5.5.1.1 User-group tokens
- section 5.5.1.3 Token for statusless logons
- section 5.10 Cover retrieval

3.3.3. Media file users

The following sections of this document are relevant for users who need to access all media files:

- sections 1 Contents through 5.3 Field content in return formats
- section 5.5 API login
- section 5.5.1 Access token for API login
- section 5.5.1.1 User-group tokens
- section 5.5.1.3 Token for statusless logons
- section 5.9 Accessing media files

3.4. Alternative methods for retrieving data via VLB API

Instead of retrieving data via VLB API, VLP data can be accessed in ONIX format via an FTP server.

Apart from ONIX data, media data (including cover data) can also be accessed via an FTP server.

If you wish to enable access to ONIX data or media files, contact MVB customer service, whose contact information can be found in section 8 Contact information .

4. VLB API technology

VLB API is based on representational state transfer (REST) technology. This means that relative to the previously used SOAP technology, changes have been made that are advantageous from the standpoint of usage and performance.

4.1. General API features

The general features of VLB API are as follows:

- The VLB interface uses the **Oauth2 authentication method**, which provides a streamlined authentication procedure that makes use of a token for further communication.
- Each data record contains a **unique 32-character alphanumeric ID** GUID (Globally Unique Identifier).
- Under this regime, two different data formats are available: **JSON (Java Script Object Notation)** and ONIX 3.0.
- JSON comes in a **JSON-wrapper** that returns the total number of results per query, the total number of pages, the number of results per page, and the page number.
- The interface handles errors via **standard HTML status and error codes**. Errors are discussed in greater detail later in this document.
- In the interest of making it easier to deal with the various builds, VLB REST API incorporates version information into the relevant URLs. This enables existing applications to remain in use without change, including in the event of an API update for a transitional phase.
- API can be used for **single-user solutions**, as well as for **webshops**. The basic configurations are implemented by MVB.
-

4.2. Data access functionalities

VLB REST API functionalities, which are similar to those found in the VLB API forerunner, are as follows:

- LogIn / LogOut
- Various search methods (quick search, Boolean search, hierarchy and series search)
- Returns in JSON, ONIX 2.1 and ONIX 3.0 formats
- Results lists in two different field composites
- Article information can be retrieved via an ID
- Article cover URLs can be retrieved
- URLs for various media files can be retrieved
- Publisher data can be accessed via publisher IDs
- Indexes for various fields can be accessed

4.3. Interface changes relative to previous interfaces

4.3.1. Changes relative to v1 of the REST interface (HTTPS advisory)

A number of changes have been made relative to REST API v1.

- Fields for which no value is available are not displayed. In other words, “zero” value fields are no longer shown.
- The json-short format is no longer provided in detail calls. In such cases, only the JSON format (json long) is provided.
- The explicit reference price indicator has been eliminated. Instead, fields needed for reference price determination are now provided.
- The interface uses native HTTPS. HTTP calls are relayed to HTTPS via status code 308.

4.3.2. Reference price display in API

Any fixed price for which no provisional price is indicated counts as a reference price. The explicit reference price indicator in a separate field has been removed from REST API version 2.

The fields containing both fixed and provisional prices can be found in the results lists and in the detailed-product information.

5. Interface calling and technical integration

5.1. Communication with VLB API

VLB REST API is called via URLs, on whose servers the relevant functions and parameters need to be defined. Each request header must contain the relevant access token, and for product queries it is necessary to indicate the return format JSON (long) or ONIX format (short, ref)].

General URL format:

<https://api.vlb.de/api/v2/<function>?<parameter>>

Calls need to be encrypted via HTTPS (see the HTTPS note in section 4.3.1 Changes relative to v1 of the REST interface (HTTPS advisory)). In the case of changes in function calls that are incompatible with the previous version, the version number (in v1 above, for example) is incremented (to v2, v3, and so on). This applies

in particular to field deletions or type modifications that need to be updated in the client system.

Every function call can return errors, the codes for which can be found in section 5.4 Errors ; other API responses can be found in section 6.3 API returns. In addition, spoken error messages can be found in JSON.

Unless otherwise indicated, an HTTP GET is used for all of the calls/requests referred to below.

5.1.1. URL live environment

The API live environment for productive operation is available via the following URL:

<https://api.vlb.de/api/v2/<function>?<parameter>>

The live environment will be used for the customer software development phase, as well as for productive use.

5.1.2. URL test environment

In certain cases, new functions and testing tools are provided via a test system. API customers are sent separate notifications in any such case. This test system is not always available in tandem with the live system. The test system need not always be identical to the live system and is subject to change.

Basic URL test system (via HTTPS): <https://test-api.vlb.de/>

5.2. Return formats

The HTTP content-type header can be used to specify the format in which the parameters are relayed. It is **absolutely essential** that the “accept” parameter be used to define the return format. The specific format that a given endpoint supports is indicated in the descriptions of the endpoint in question.

Return formats:

- *Application/json-short* is used to return the “JSON-short” format (*applies to results lists only*)
- *Application/json* is used to return the “JSON-long ” format (*applies to results lists and detail calls only*)
- *application/onix21-short* returns ONIX21 in the short format (for detail calls only)
- *application/onix21-ref* returns ONIX21 in the reference format (for detail calls only)
- *application/onix30-short* returns ONIX3.0 in the short format (for detail calls only)

- *application/onix39-ref* returns ONIX3.0 in the reference format (for detail calls only)

5.3. Field content in return formats

Fields containing no values are not output. Thus the scope of a given return format is determined by the content of the data record in question. On the other hand, previous interface version v1 output ZERO values in cases where no data was available.

5.4. Errors

5.4.1. Error codes

API returns errors as http status codes, as follows:

- 400 Bad Request: The request was not structured properly.
- 401 Unauthorized: Cannot be carried out without valid authorization.
- 403 Forbidden: Request not allowed in the form provided.
- 404 Not Found: The resource could not be found.
- 500 Internal Server Error: For general status errors

If archived titles are requested for archived titles without authorization, the *401 – Unauthorized* message is displayed because the token lacks authorization to retrieve the archived data.

5.4.2. JSON error codes

Apart from the HTTP error codes, additional error information is provided in the body of the responses, in JSON format. These messages are structured as follows:

```
{
  "error": "<error_code>",
  "error_description": "<error_description>"
}
```

The possible error codes are as follows:

- *invalid_token*: The transferred token is invalid.
- *product_blocked*: The requested product is a read-only item (i.e. for display only).
- *no_permission*: This request cannot be fulfilled with the permissions level of the token provided.
- *not_found*: Product unavailable
- *unauthorized*:
- *access_denied*:

For further information on error messages, see section 6.3 API returns.

5.4.3. restrictions for API request (Deep paging lock-up)

There is a limit of 10.000 articles to be requested by an API call to avoid deep paging.

The hitlist can contain more articles but it is not possible to go by the page and size parameters above a limit of 10.000 articles. If the combination of page and size reaches 10.000 an error message is shown.

For example size 250 and page 40 will be a number of 10.000 articles. There the last article in the list on page 40 can still be shown. But calling page 41 will result in a list which is beginning with article nr 10.001 which is above the limit and will result in an error message. The same error message will be shown when calling a size of 100 and the page nr 101 which will result in a hitlist starting with article nr 10.001 which is above the limit (see chapter 6.3.2.1 Responses to search calls).

The API request can still show a number of “totalElements” which is higher than 10.000 but the article metadata above 10.000 can be requested.

The deep paging lock-up does not affect article call in detail by ISBN or UUID as in this case only one article is requested with each call.

To avoid the deep paging lock-up use the Boolean query parameters to adapt the API request.

5.5. API login

Owing to the use of REST technology and the OAUTH2 procedure, API uses an access token for authentication. This token needs to be relayed for each request that is sent to API.

5.5.1. Access token for API login

Depending on the application, a distinction is made between a status-based logon for single-user applications and a statusless logon for webshop applications. The corresponding tokens are provided or generated.

5.5.1.1. User-group tokens

Depending on user group and the attendant access permissions, a given token enables access to the following:

- Metadata and media objects, including cover data
- Media objects only, including cover data
- Cover data only
- Metadata only

- Address data

5.5.1.2. Token for status-based logons

For single-user applications, the requisite access token is generated via the login. Such tokens are only valid for logins for the particular application in question

A login-password combination enables a limited number of parallel logins, i.e. an access token is generated in each such case. Once this numerical limit has been reached, an access token must be “returned” (i.e. released) before a new token can be generated. The number of allowable parallel API logins for such a user/account is determined by MVP customer service.

For scenarios involving a single-user application, API has a defined timeout of 60 minutes for inactivity, whereupon the token’s status is hierarchy to “invalid.” During the first ten minutes of a given token’s validity, API use does not trigger an extension of the token.

5.5.1.3. Token for statusless logons

For webshop applications or applications with an unlimited number of data users, an access token is provided that enables an unlimited number of parallel applications. No advance login is required in such cases. Access to media objects is subject to restriction depending on the permissions in question (customer-account billing). In such cases, tokens are made available via MVB customer service. To retrieve address data, a function other than article metadata must be used.

5.5.2. Authentication

For all requests (except logins), access is authenticated using an access token, which can be provided in two ways:

1. HTTP authorization header with the access token “bearer” value:

<https://api.vlb.de/api/v2/product/29ef901373fb48bea8865948c75af64f>

Antwort-Header
Connection close
Content-Type application/json; charset=UTF-8
Date Fri, 16 May 2014 07:41:43 GMT
Transfer-Encoding chunked
Anfrage-Header
Accept application/json, text/javascript, */*; q=0.01
Accept-Encoding gzip, deflate
Accept-Language de, en-US;q=0.7, en;q=0.3
Authorization Bearer 106dbc7d-2295-4518-a16b-6567c31eb80e

- URL Parameter access_token:

https://api.vlb.de/api/v2/logout?access_token=98592fc6-61e6-4ed2-8f9e-f3fe7700bd79

If an authentication fails, http status code 401 is returned in JSON format:

```
{  
  "error": "invalid_token",  
  "error_description": "Invalid access token: 98592fc6-61e6-4ed2-8f9e-f3fe7700bd79"  
}
```

5.5.3. Status-based login

5.5.3.1. Login

Login is required for use in enterprise resource planning systems or single-user search systems, among others. In such cases, the user needs to have a login-password combination in order to log in to API. Once the user has logged in, the system generates an access token.

Login request

The user is logged in and is given a valid access token.

URL: The login functionality that API uses to generate access tokens is called using the following URL/function:

Live: <https://api.vlb.de/api/v2/login>

Procedure:

HTTPS

Method:

POST

Authentication required: No

Request Parameter:

None

Request Payload:

JSON with the following attributes:

- username: your username
- password: your password

Example:

```
{  
  "username": "your username",  
  "password": "your password"  
}
```

Return:

HTTP-StatusCode: 200

Format: String

Content: accessToken

5.5.3.2. Logout

The access token relayed with the request is logged off by the system. Also see section 5.5.2 Authentication.

URL:

<https://api.vlb.de/api/v2/logout>

Procedure:

HTTPS

Authentication required:

Yes

Request Parameter

None

Return

HTTP-StatusCode: 200

Format: None

Content: No

5.5.4. Requests for media objects

For security reasons, a separate access token must be obtained in order to request cover/media file data. Such tokens need to be generated as described above (see section 5.5.1.3 Token for statusless logons).

5.5.5. Notes on access token use

Although these tokens enable access to the VLB system without further authentication, cover data and media objects tokens cannot be used to retrieve metadata. It must be ensured that access tokens cannot be copied or viewed by third parties via a proxy or by any other means.

5.5.6. Description

VLB API provides a range of search tools that are also available from VLB online at www.vlb.de:

- Quick search
- Boolean search
- Batch search
- Index search (see section 5.11 Index searches)

As a rule, stop words are excluded from searches. These words include the following, among others: a, al, das, dem, den, der, die, ein, eine, einem, einen, einer, eines, el, I, la, le, les, lo, the, un, una, une, uno.

5.6. Searches

5.6.1. Quick search

Quick searches bring the following fields into play:

- Keywords (i.e. assigned by the publisher, or DNB (German National Library) keywords); title and subtitle
- Series and hierarchy titles
- Author
- Publisher
- Identifier (ISBN, GTIN, ISSN, ÖSB-number, VLB order number, etc.)
- Subject
- Product-group index name
- Product-group code name (e.g. narrative literature)
- ZIS topic-group name (e.g. chemical industry, plastics)
- Series abbreviationX number for sets

5.6.2. Boolean searches

Boolean searches use standard VLB Boolean parameters. For further information in this regard, see the VLB online help rubric entitled.

Key	Searched indexed fields	Comment
ST	Quick-search fields 5.6.1	Keyword
AU	contributors	Author, copyright holder
SW	keywords	Keyword; searches of publisher-designated and DNB (German National Library) keywords
TI	title	Title
VL	publisher publisherid	Publisher and publisher ID no. (publisherMvBld, mvBld)
IS	isbn gtin oesbnr Order no.	All possible identifiers. Note: only VLB order numbers are searchable, to the exclusion of delivery or book wholesaler number
PR	currentPriceDe	Price
EJ	publicationDate	Year of publication
WG	genreCode	Product group no.: preceded by product group index; truncating allowed (see section 5.6.2.2 Important advisory concerning Boolean searches:)
SP	language	Language

Key	Searched indexed fields	Comment
AD	lastModificationDate	Last updated; entry: yyyyymmdd
ZD	creationDate	Access date: entry: yyyyymmdd
TH	THEMA_MAIN SUBJECT, THEMA_MAIN SUBJECT_NAME, THEMA SUBJECT, THEMA SUBJECT_NAME, THEMA_QUALIFIER, THEMA_QUALIFIER_NAME,	Topic codes: Here you can search for topics and qualifiers as well as truncated versions thereof (e.g. TH=A finds all titles classified under "A"; TH=A* finds all titles that have a topic classification starting with A, i.e. AB, AF, AG; AFC, AFH etc.; this applies to qualifiers as well)
TM	THEMA_MAIN SUBJECT, THEMA_MAIN SUBJECT_NAME	Thema-Main-Subjects (Subjects)
TS	THEMA SUBJECT, THEMA SUBJECT_NAME	Thema-Subjects (Subjects)
TQ	THEMA_QUALIFIER, THEMA_QUALIFIER_NAME	Thema-qualifier (Qualifier)
PF	productForm	Product form (ONIX codes); truncating allowed (PF=E*) returns all digital products) Please note that certain valid product-form codes have been added to Onix 3.0. ONIX codes – List 150; truncating is allowed Onix 2.1 product forms (PF) and Onix 3.0 product form details (PD) cannot be combined and are not to be used.
PD	productFormDetail	ONIX codes – List 175; truncating is allowed
PT	productType	These Boolean parameters can be used to select products with VAT splitting in a special bundle. The following parameter is case sensitive: a.:multi-Bundles PT=multiBundle b.:duo-Bundles PT=duoBundle c.:all VAT splitting Bundles PT=*>Bundle Also see section 9.4.1 productType values
RH	Series/Hierarchy Title, Series/Hierarchy Identifier	The key applies to series such as hierarchy searches. Series

Key	Searched indexed fields	Comment
		and hierarchy identifiers are differentiated by the first character:
LI	Availability	Publisher availability information from j396
UG		Copyright GND (no wildcard searches)
UI		Copyright ISNI (no wildcard searches)
UO		Copyright ORCID (no wildcard searches)
LH	Main-reading-rational (Hauptlesemotiv)	ONIX 3.0 Codelist 27 , Code B8 (siehe 5.6.2.3 Reading-rationales (Lesemotive) in the Boolean search
LN	Sub-reading-rational (Nebenlesemotiv)	ONIX 3.0 Codelist 27, Code B8 (siehe 5.6.2.3)

Categories can be entered in any combination of upper and lower case; an equals sign must be inserted right after the category without a line space.

The category entry for the first category may be omitted, in which case ST will be selected by default.

Relative to the previous interface version v1, the parameters PD (product form detail) and LI (availability) have been added as well as the parameter PF (product form), which searches the product form values of the ONIX 3.0 product form.

5.6.2.1. *Search syntax and Boolean operators*

Once a **category has been entered**, you may enter one or more search terms, separated by line spaces. If multiple search terms are entered, they are linked by default with AND (in the same field of the category, but not necessarily in the same line, in cases where a field can have multiple values).

Example: au=Karl May *finds all works whose authors, publishers and so on contain at least one instance each of “Karl” and “May” – and of course any instance of “Karl May.”*

Diacritical accents need not be entered and are normalized automatically. **These searches are not case sensitive. German umlautes** are automatically set to their multi-letter equivalents (ü <-> ue, ß <-> ss etc.).

If you put quotation marks around a search phrase, it will be searched in precisely this order (phrase).

Example: au="May, Karl" *finds all works by Karl May, as names are saved in this form.*

If multiple categories are entered (which must be separated by line spaces), they are automatically linked by AND.

Example: ti=gymnastik pf=BC sp=ger *finds all German-language paperbacks with the word "gymnastik" in the German-language title*

Alternatively, categories can also be explicitly linked using AND/OR and can be parenthesized if necessary.

Example:

- ti=gymnastik or ti=turnen finds all works with "gymnastik" or "turnen" in the title
- au=rowling and (pf=AJ or pf=EA) finds all works with "rowling" as an author that are either downloadable audio files or e-books.
- st=water and (PF=not EA or PF=BA)

You are not allowed to insert operators within a search phrase. For example, ti=gymnastik or turnen" does yield the desired results, it must then be executed as in the above example, with multiple category naming.

To exclude a particular term from a search, insert NOT in front of it. This will result in all works being searched that do not contain the term in question.

Example: ti= not fish not meat *finds all works whose titles do not contain the words meat or fish.*

The sole allowable Boolean operators (for German and English) are **UND/AND**, **ODER/OR** and **NICHT/NOT**. Use of symbols for these operators (&, | , -) is *not* allowed. To search for terms or phrases containing one or more of these operators, they must be enclosed in quotation marks (ASCII 22) in order not to be interpreted as operators.

Example: ti="not fish not meat" *finds all works in which the phrase "not meat not fish" occurs.*

Use the caret symbol (^) to search for periods of time. An example of such a search would be search=AD=20150319^20150321, whereby both the start and end dates are inclusive. Please always use the URL encoding when generating the boolean search phrase (for example the search pharse above: AD%3D20150319%5E20150321)

Date categories can be 4 digits (YYYY), 6 digits (YYYYMM) or 8 digits, in accordance with the ONIX date format (YYYYMMDD). Alternatively, the German date format DD.MM.YYYY can be used. The latter format can only be used in its entirety, i.e. all 8 digits. Wildcard searches cannot be carried out with the German date format. The ONIX date format enables solely 4 or 6 character wildcard searches.

5.6.2.2. *Important advisory concerning Boolean searches:*

When searching using Boolean parameters, it is essential to keep certain considerations in mind for some fields.

- Product group searches
 - Searches in the product-group index: WG=1*
(here, searches for hardcover and paperback titles – WG Index 1)
 - Searches according to product group: WG=*250
(here, searches for product group 250, i.e. books for children age 11 or under)
- Topic searches:
 - For example, th=la finds all titles with topic code la; th=la* also finds "la" in topic texts on topics such as German regional-state and local politics.
- Special symbols
 - If quotation marks are used in a search, the ASCII code 22 must be used, e.g. "Hello out there"
- Product form and product form details
 - Onix 3.0 enables the use of Onix 3.0 product form codes. In other words, e-book titles can now be searched via PD=E101; plus searches of categories of electronic products are now enabled via PF=EA and the like.
 - Product form codes from ONIX 2.1 cannot be combined with codes from ONIX 3.0, i.e. PF=DG and PD=E101.

5.6.2.3. *Reading-rationales (Lesemotive) in the Boolean search*

The reading-rationales (Lesemotive) could be used in the Boolean search as any other classification (ONIX definition 9.5.23 subjects – product classifications).

Each title receives an automatically generated Main-reading-rational and additionally a sub-regarding-rational can be generated (excluded are childrens books up to 11 years and present books).

The reading-rationales are only available within the API detail call. The API hitlist does not show the reading rationales).

If searching the reading-rationales for a single publisher or for a single publisher who is part of a publisher group the single publisher-ID should be used (no or / oder combination of the publisher-IDs of a publisher group).

The actual reading-rationales can be checked on the VLB help pages using the URL:
<https://vlb.de/hilfe/lesemotive>

See also the table listing all Boolean search parameters at the beginning of this chapter 5.6.2 Boolean searches.

5.6.3. Batch searches

A list of up to 500 ISBNs can be sent to the batch search function. Batch searches can be combined with Boolean parameters.

Use an HTTP POST for batch searches. ISBNs can be entered in Payload, as follows:

```
{  
  "content": [  
    {  
      "isbn": <value>,  
    },  
    {  
      "isbn": <value>,  
    },  
  ]  
}
```

5.6.4. Request

URL:

[https://api.vlb.de/api/v2/products?page%3D<pageno>%26size%3D<size>%26\[sort%3D<column>%26direction%3D<direction>\]\[%26search%3D<value>\]\[%26active%3D<active>\]\[%26source%3D<source>\]](https://api.vlb.de/api/v2/products?page%3D<pageno>%26size%3D<size>%26[sort%3D<column>%26direction%3D<direction>][%26search%3D<value>][%26active%3D<active>][%26source%3D<source>])

Procedure:

HTTPS

Method:

POST

Authentication required:

Yes

Request Parameter:

- page: the page number of the page to be output (begins with 1)
- size: the number of products per page that is to be output. The maximum number of products per page is 250.
- sort (optional): The output column in which the search results are to be sorted. Default sorting (“sort” parameter not specified) is output by descending score (value of the title to the search). The following columns are possible:
- identifier
 - author
 - titleAndSubtitle
 - publisher
 - publisherMvId
 - publicationDate
 - productAvailability
 - price
 - creationDate
 - lastModificationDate
 - productType
 - active

direction (optional): sort in ascending or descending order. The allowable values are **asc** (ascending) and **desc** (descending).

search (optional): a (Boolean) search phrase (with encoded URL; see section 7. Encoding). If not specified, all available products will be searched. Fields can be explicitly searched via Boolean parameters. In cases where the search phrase contains no limiting search fields, the system will search in the quick search fields.

active (optional): active (**true**) and/or inactive (**false**) title
(if not indicated: active and inactive titles)

source (optional): defines the aggregator of the data to be provided. Can be linked at will. For further information in this regard, see section 5.13 Aggregators (overview)

Request Payload:

JSON with the following attributes:

- ISBN

5.6.4.1. *Return – results list*

HTTP status code: 200

Format: JSON (short or long)

Content:

content: products from search results (basic data).

The key fields in this regard are as follows:

productId: Unique (internal) product ID

version: Unique version number of the product (for in-house use)

identifier: The main identifier of the product – normally ISBN13. Can occur multiple times per selection. The main identifier is unique for each aggregator.

sort: Results-list sorting

direction: Ascending (asc) or descending (desc)

property: The results column that formed the basis for sorting

ascending: True, if “asc,” otherwise false

totalPages: Number of search-results pages

lastPage: True, if this page is the last search-results page

numberOfElements: Number of results shown on this page

totalElements: Total number of results

firstPage: True, if this page is the first search-results page

size: Maximum number of products requested

number: Current page number. Begins with 0 – and thus corresponds to the page - 1 search parameter

JSON-short return:

```
{
  "content": [
    {
      "productId": "1f53c4e19dda4d4499718a75f81f8577",
      "isbn": "978-3-7657-6000-6",
      "gtin": "9783765760006",
      "order no.": "1876160",
      "author": "Corporation is author; Brown, Dan"
      "title": "Hierarchy title / hierarchy title / (main) title;"
      "subTitle": "hierarchy subtitle / subtitle";
      "publisher": "Reclam, Philipp",
      "publisherMvbId": "5108746",
      "publicationDate": "10.10.2021",
      "edition": "Edition text: second expanded and annotated edition, version 3",
      "productType": "duoBundle",
      "priceEurD": 22.6,
      "collections": [
        {
          "setId": "X-3527-6306-X",
          "title": "Hierarchy title"
          "subtitle": "hierarchy subtitle"
          "partNumber": 2
        },
        {
          "collectionId": AADUN86
          "title": "Series title"
          "subtitle": "Series subtitle"
          "partNumber": 5
        }
      ],
      "announcementDate": "03.09.2020",
      "productForm": "SA",
      "primaryContentType": "10",
      "productFormDetail": [
        "B131",
        "E200",
        "B105",
        "V221"
      ],
      "priceProvisionalEurD": false,
      "priceCalculatedEurD": false,
      "priceFixedEurD": true,
      "active": true,
      "productAvailability": "10",
      "creationDate": "28.06.2018",
      "lastModificationDate": "28.06.2018"
    }
  ],
  "totalElements": 1,
  "totalPages": 1,
  "last": true,
  "firstPage": true,
  "lastPage": true,
  "numberOfElements": 1,
  "sort": [
    {
      "direction": "ASC",
      "property": "identifier",
      "ignoreCase": false,
      "ascending": true
    }
  ],
  "first": true,
  "size": 250,
  "number": 0
}
```

5.7. Retrieving lists of product entries

The product-entry list retrieval tool enables you to readily access detailed information on various products.

To retrieve the data, enter the UUID of the desired products. The list of product entries in the response corresponds to the order of the requested UUIDs.

Use an HTTP POST for batch searches. UUIDs can be entered in Payload, as follows:

```
{  
  "ids": [  
    <value>,  
    <value>,  
  ]  
}
```

5.7.1. Request

URL:

<https://api.vlb.de/api/v2/product/multipleProducts>

Procedure:

HTTPS

Method:

POST

Authentication required:

Yes

Request Parameter:

None

Request Payload:

JSON with the following attributes:

Product ID

5.7.2. Return: list of product entries

HTTP status code: 200

Format: String

Content:

In the following, only the key parameters will be discussed:

productId: Unique ID of a data record. Each sub-entity provides an ID that is a 32-digit hash value.

version: The version containing the data record. The version number is incremented each time a change is made.

composites: refers to the individual sections (audiences, contributors, etc.) These usually depend on the ONIX structure and contain speaking parameters

Returns are provided in JSON format only. Returns in such cases cannot be provided in JSON-short or ONIX.

5.8. Product-entry retrieval

Detailed product entries can be retrieved via the 32 digit product ID in cases where the information provided by the results list is insufficient. Two output formats (JSON-long and ONIX) are available.

5.8.1. Product entry returns

HTTP status code: 200

Format: String

Content:

In the following, only the key parameters will be discussed:

productId: Unique ID of a data record. Each sub-entity provides an ID that is a 32-digit hash value.

version: The version containing the data record. The version number is incremented each time a change is made.

composites: refers to the individual sections (audiences, contributors, etc.) These usually depend on the ONIX structure and contain speaking parameters

Returns for individual product calls are provided in two different formats: JSON and ONIX. Two different output profiles are available for the JSON format.

In the ONIX format, only data from VLB is available, i.e. data from other aggregators cannot be retrieved in this format.

5.8.1.1. JSON-long

The JSON-long format contains all metadata and all full-length long-text fields for a given product. It should be noted that long texts in VLB can be considerably longer, as there are no limits on their character counts.

The “publisherData” block is not available for statusless tokens (5.5.1.3 Token for statusless logons).

5.8.1.2. ONIX30 short

To view the exact return parameters in ONIX 3.0 format, see the relevant ONIX specifications.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ONIXmessage release="3.0">
<header>
  <sender>
    <x298>VLB - Staging</x298>
    <x299>MVB Marketing- und Verlagsservice des Buchhandels GmbH, 069.1306.550</x299>
    <j272>serviceline@mvb-online.de</j272>
  </sender>
  <x307>20180709</x307>
</header>
<product datestamp="20180706">
  <a001>392484a6cc2d4db4a165ec94baac234c</a001>
  <a002>03</a002>
  <productidentifier>
    <b221>02</b221>
...
</OnixMessage>
```

5.8.1.3. ONIX30 ref

To view the exact return parameters in ONIX 3.0 reference format, see the relevant ONIX specification.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ONIXMessage release="3.0">
<Header>
  <Sender>
    <SenderName>VLB - Staging</SenderName>
    <ContactName>MVB Marketing- und Verlagsservice des Buchhandels GmbH,
069.1306.550</ContactName>
    <EmailAddress>serviceline@mvb-online.de</EmailAddress>
  </Sender>
  <SentDateTime>20180709</SentDateTime>
</Header>
<Product datestamp="20180706">
  <RecordReference>acf1e447f5c1489b9b65fae5ee5a4fa7</RecordReference>
  <NotificationType>03</NotificationType>
  <ProductIdentifier>
    <ProductIDType>02</ProductIDType>
...
</OnixMessage>
```

5.8.2. Request

URL:

[https://api.vlb.de/api/v2/product/<id>\[/<id-type>\]](https://api.vlb.de/api/v2/product/<id>[/<id-type>])

Procedure:

HTTPS

Authentication required:

Yes

Request Parameter

id: The ID of the product that is to be displayed. This ID is shown in the field ID search results, from which it can be referenced. is also shown in the “identifier” field for id-type

optional ID types: The type of retrieved ID. If none is indicated, the unique ID is used as the standard ID. Possible values:

- gtin
- isbn13
- ean

5.8.3. Example of product-entry retrieval

A product is requested in JSON-long format:

<https://api.vlb.de/api/v2/product/79b7ee6ef4e74f4f9cbb242a6e30e537>

or

a produkt is requested in JSON-long Format using ISBN (id-type):

<https://api.vlb.de/api/v2/product/9783123456789/isbn13>

Results:

See the hyperlinked ZIP file in section 6.4 Specimen data.

5.9. Accessing media files

5.9.1. Description

Apart from cover data URLs, VLB API provides URLs for the media files of a given article. This includes links to back covers, inside views, etc.

The endpoint of the request is the same as in the previous version of the interface (v1) – which means that media and cover data can be retrieved via either a v1 or a v2 URL. When you request media or cover data, bear in mind that you need specific authorization in order to use access tokens – copying or viewing of which by third parties is strictly prohibited. Thus you will need to choose at least one authentication method that precludes reading of the token in question directly from the media file URL. For further information concerning the various methods that come into play here, see section 5.5.2 Authentication. Alternative retrieval methods for media file URLs, such as a proxy, can also be used.

Token for metadata searches and requests (i.e. dedicated metadata tokens) cannot be used to retrieve cover data or media files.

The following media files can be accessed via VLB REST API.

Media file type jpg, png: It depends which kind of media file type the publisher provided to the vlb. If png files are provided, png files will be distributed. If jpg files are provided then jpg files are distributed.

Name	JSON type	Media type	File type:
Cover	FRONTCOVER	Image file	Jpg, png
Backcover	BACKCOVER	Image file	Jpg, png
Other cover data	COVER_PACK	Image file	Jpg, png
Inside view	IMAGE_SAMPLE_CONTENT	Image file	Jpg, png
Table of contents:	TABLE_OF_CONTENT	PDF file	Pdf
Description	DESCRIPTION	PDF file	Pdf
Author photo	AUTHOR_IMAGE	Image file	Jpg, png
Audio sample	AUDIO_SAMPLE_CONTENT	Audio file	mp3, wave
Text sample	TEXT_SAMPLE_CONTENT	PDF/e-pub file	pdf, epub
Review	REVIEW	PDF file	pdf
Foreword	INTRODUCTION	PDF file	pdf
Index	PRODUCT_INDEX	PDF file	pdf
Publisher logo	PUBLISHER_LOGO	Image file	Jpg, png
Imprint logo	IMPRINT_LOGO	Image file	Jpg, png
Author presentation (text)	AUTHOR_PRESENTATION_TEXT	PDF file	Pdf
Author presentation (audio)	AUTHOR_PRESENTATION_AUDIO	Audio file	mp3, wave
Author interview (text)	AUTHOR_INTERVIEW_TEXT	PDF file	Pdf
Author interview (audio)	AUTHOR_INTERVIEW_AUDIO	Audio file	mp3, wave
Author reading from his work	AUTHOR_READING	Audio file	mp3, wave
Stage image	STAGE_IMAGE	Image file	Jpg, png
Feature	FEATURE_ARTICLE	PDF file	Pdf
Press release	PRESS_RELEASE	PDF file	Pdf
Promotional material (text)	PROMOTIONAL_EVENT_MATERIAL_TEXT	PDF file	Pdf
Promotional material (audio)	PROMOTIONAL_EVENT_MATERIAL_AUDIO	Audio file	mp3, wave
Promotional material (image)	PROMOTIONAL_EVENT_MATERIAL_IMAGE	Image file	Jpg, png
Tutorial (text)	INSTRUCTIONAL_MATERIAL_TEXT	PDF file	pdf

Name	JSON type	Media type	File type:
Tutorial (audio)	INSTRUCTIONAL_MATERIAL_AUDIO	Audio file	mp3, wave
Tutorial (image)	INSTRUCTIONAL_MATERIAL_IMAGE	Image file	Jpg, png
Marketing graphic	ONLINE_ADVERTISEMENT_PANEL	Image file	Jpg, png
Building or game instructions	MANUAL	PDF-Datei	Pdf
Errata	ERRATA	PDF file	pdf
Series description	COLLECTION_DESCRIPTION	PDF file	pdf
Series image	SERIES_IMAGE	Image file	Jpg, png
Series logo	SERIES_LOGO	Image file	Jpg, png
Bibliography	BIBLIOGRAPHIC	PDF file	pdf
Preview	PUBLISHERS_CATALOGUE	PDF file	pdf
Product image	PRODUCT_ARTWORK	Image file	Jpg, png
Product logo	PRODUCT_LOGO	Image file	Jpg, png
Wallpaper	WALLPAPER	Image file	Jpg, png
Full Cover	FULL_COVER	Image file	Jpg, png
Brand logo	MASTER_BRAND_LOGO	Image file	Jpg, png

5.9.2. Request

URL:

<https://api.vlb.de/api/v2/asset/mmo/<productId>>

Procedure:

HTTPS

Authentication required:

Yes

Request Parameter

Product ID: Unique identifier of a given product whose media file URLs are to be retrieved

5.9.3. Return

HTTP status code: 200

Format: JSON

Content:

Type: Media file type. The possible types are listed in section

5.9.1 Description (example):

- Cover
- Backcover
- (see table)

url: The actual URL from which a given media file can be downloaded after authentication.

```
[  
 {  
   "type": "<type>",  
   "url": "<url>",  
   "sequenceNumber": "<number>"  
 }  
]
```

5.9.4. Example

All media files for a given product are displayed.

Example for EPUB and front covers

<https://api.vlb.de/api/v2/asset/mmo/192a806ee0bf483da8fad6b7b8ec9308>

plus an example for other media types

<https://api.vlb.de/api/v2/asset/mmo/518498f30da14eaab8d34364ee2610a8>

Results:

```
[
  {
    "type": "FRONTCOVER",
    "url": "https://api.vlb.de/api/v2/cover/9783150191064/m",
    "sequenceNumber": 1
  },
  {
    "type": "TABLE_OF_CONTENT",
    "url": "https://api.vlb.de/api/v2/asset/mmo/file/bfd3f377-4eaa-4b65-b830-577918d776c8",
    "sequenceNumber": 1
  },
  {
    "type": "TABLE_OF_CONTENT",
    "url": "https://api.vlb.de/api/v2/asset/mmo/file/6adc9faa-9bb1-46c0-80f1-01e6b0739ce9",
    "sequenceNumber": 2
  },
  {
    "type": "PREVIEW_SAMPLE",
    "url": "https://api.vlb.de/api/v2/asset/mmo/file/9c350d33-720d-477b-87ee-ea07b39e2396",
    "sequenceNumber": 1
  },
  {
    "type": "PREVIEW_SAMPLE",
    "url": "https://api.vlb.de/api/v2/asset/mmo/file/5e7d4ba7-b121-45ff-a85e-f85a831459b5",
    "sequenceNumber": 2
  }
]
```

5.10. Cover retrieval

5.10.1. Description

The cover for a given title can be retrieved in various sizes via the referenced coverlink. To identify a cover, use the ISBN/GTIN of the relevant title.

The endpoint of the request is the same as in the previous version of the interface (v1) – which means that media and cover data can be retrieved via either a v1 or a v2 URL.

But unlike in this function, cover URL results lists display the cover in size M.

When retrieving cover objects, bear in mind that you need specific authorization in order to use access tokens – copying or viewing of which by third parties is strictly prohibited. Thus you will need to choose at least one authentication method that precludes reading of the token in question directly from the cover URL. For further information concerning the various methods that come into play here, see section 5.5.2 Authentication. Alternative retrieval methods for cover URLs, such as a proxy, can also be used.

Tokens for metadata searches and requests cannot be used to retrieve cover or media files.

5.10.2. *Cover-Request*

URL:

[https://api.vlb.de/api/v2/cover/<id>/\[<size>\]](https://api.vlb.de/api/v2/cover/<id>/[<size>])

Procedure:

HTTPS

Authentication required:

Yes

Request Parameter:

id: The product or cover GTIN is requested (not hyphenated – i.e. like this for example: 9783411046508)

size (optional): cover size. In the software's current build, three values can be relayed (the parameter is case sensitive):

- s (small)
- m (medium)
- l (large)
- No parameter = original size

Size	Call parameter	Width	Height	Comments
Small	s	90	xyz	Height proportional to width
Medium	m	200	xyz	Height proportional to width
Large	l	abc	599	Width proportional to height
(Original size)*		abc	xyz	Original cover size in VLB

*Note: “Original size” refers not to the file size, but rather to the original pixel size. Covers in the original size are provided in a web-optimized format.

5.10.3. *Return*

HTTP status code: 200

Format: image/jpeg

Content: Cover in binary form

5.11. Index searches

5.11.1. *Description*

Searches of certain fields can be facilitated by calling a search index. In such cases, a search is effected for the next 100 entries for a particular word or term and a results list is generated.

Index searches for the following fields are enabled:

- Author
- Publisher
- Title
- Keyword
- Series
- Hierarchy
- Identifier (ISBN / EAN, Austrian school textbooks etc.)

Scrolling is not supported in the index retrieval function. Hence you need to either limit or expand the scope of index search terms.

5.11.2. *Request*

URL:

<https://api.vlb.de/api/v2/index/<field>/<search-term>>

Procedure:

HTTPS

Authentication required:

Yes

Request Parameter

field: the field that is to be searched. This can be one of the following:

- author
- publisher
- title
- keyword
- set
- collection
- identifier (ISBN / EAN)

search-term: the actual search term

5.11.3. *Return*

HTTP status code: 200

Format: Json

Content:

content: contains a list of all results

value: value of the results

count: the frequency of the occurrence of the results value referred to above.

```
[  
 {  
   "value": <value>,  
   "count": <value>  
 }  
]
```

5.11.4. *Example*

A user searches for an author by the name of “Meier.”

<https://api.vlb.de/api/v2/index/author/meier>

Results:

```
[  
 {  
   "value": "Meier",  
   "count": 5  
 },  
 {  
   "value": "Meier, Achim",  
   "count": 2  
 },  
 {  
   "value": "Meier, Adolf",  
   "count": 1  
 }  
]
```

5.12. Accessing publisher data

5.12.1. *Description*

This function function enables the retrieval of detailed publisher data. Such data is derived from the AdB-Online database and the [address book of the German-speaking book trade](#). Such data is accessible by subscription only.

Publisher data is retrieved via the relevant publisher ID (field publisherMvblId from the hitlist), which is relayed in a title for each publisher name.

Please regard the latest changes in autumn 2024, which are shown in chapter 5.12.2 Return and 5.12.3 Example.

5.12.1. Request

URL:

<https://api.vlb.de/api/v2/publisher/<mvbid>>

Procedure:

HTTPS

Authentication required:

Yes

Request Parameter

id: The ID of the publisher for which data is to be accessed.

5.12.2. Return

HTTP status code: 200

Format: Json

Content:

The key return parameters are as follows:

mvbld:	Publisher's MVB ID (same as request parameter)
shortName:	Publisher's short name
name:	Publisher's full name
cityStreet:	City/town
zipStreet:	Postal code
street:	streername
street2:	additional address info
country:	Country:
postbox:	Publisher's PO box
zipPostalBox:	Postal code of publisher's PO box
phone:	Phone number
fax:	Fax
email:	Mailing address
url:	URL of publisher's website
isbnPrefixes:	All ISBN prefixes that are assigned to this publisher (as a block)
deliveryBS:	Wholesale bookseller's delivery information
deliveryNote:	SOA delivery note
bagCreditor:	Invoicing via the BAG (true/false)
bagDebitor:	Invoicing via the BAG (true/false)
tax:	Publisher's VAT ID no.
siglDe:	German logogram/official seal
siglAt:	Austrian logogram/official seal
siglCh:	Swiss logogram/official seal

vnrKreditor:	Creditor transaction no. in Germany
vnrKreditorAt:	Creditor transaction no. in Austria
vnrKreditorCh:	Creditor transaction no. in Switzerland
gln:	GLN (global location number)

5.12.3. Example

Accessing publisher data for a publisher whose identifier is 5106488:

<https://api.vlb.de/api/v2/publisher/5106488>

Results:

```
{
  "mvbId": "5106488",
  "shortName": "MVB",
  "name": "MVB Marketing- und Verlagsservice des Buchhandels GmbH",
  "country": "D",
  "zipStreet": "60311",
  "cityStreet": "Frankfurt am Main",
  "street": "Braubachstr. 16",
  "postbox": "10 04 42",
  "zipPostalBox": "60004",
  "phone": "+49 (069) 1306-550",
  "fax": "+49 (069) 1306-255",
  "email": "vertrieb@mvb-online.de",
  "url": "www.mvb-online.de",
  "isbnPrefixes": [
    "978-3-655",
    "978-3-89558",
    "978-3-9818608",
    "978-3-7657",
    "978-3-946404"
  ],
  "bagCreditor": false,
  "bagDebitor": false,
  "tax": "DE114130036",
  "vnrKreditor": "10871",
  "gln": "4280000131003"
}
```

5.13. Aggregators (overview)

In VLB REST API, data sources are referred to as aggregators. An aggregator is assigned to each title.

Aggregator	Source ID (for selection)	Aggregatorid (in the return)	Title
VLB	vlb	zero	VLB titles have no aggregators
ÖSB	oesb	5001015	Austrian school textbook data (there are no archived OESB products)
ALL	all	One ID per title	All possible aggregators combined

Aggregators can be combined by dint of multiple occurrences of the source parameter (example: ...?source=vlb&source=oesb).

Please note the following: Given the fact that no zero values are output in API v2, the "VLB" aggregator is indicated by omitting it.

6. Examples of requests and responses

6.1. Overview of API calls

Base URL: <https://api.vlb.de/api/v2/>

Function	Type:	Function calls	Parameters	Payload
Login	POST	login		Username / Passwort
Logout	GET	logout		
Searches (quick and Boolean searches)	GET	products?	Request parameters, searches	
Batch searches	POST	products?	Request Parameter	ISBNs
Index searches		index/	Index type, value	
Individual products	GET	product/	Product ID, ID type	
List of product entries	POST	product/multipleProducts		Product Ids
Publisher data	GET	publisher/	MVB-ID	
Cover	GET	cover/	ISBN, sizes	
Media objects	GET	asset/mmo/	Product ID:	

For further information, see section 5 Interface calling and technical integration

6.2. Requests and calls

This overview contains a series of sample calls, with explanations. Please note the basic calls that are needed for each of the examples, i.e. searches, article-detail call, index, publisher data and media objects.

Request type	Section	Type	Queries	Comments
Quick search	5.6.1 Quick search	GET	search=Carlos%20Ruiz%20Zaf%C3%B3n	Rapid search for the name <i>Carlos Ruiz Zafón</i>
Boolean search for a series	5.6.2 Boolean searches	GET	search=RH%3DRC712	Search by series ID
Boolean search with additional date limit	5.6.2 Boolean searches	GET	search=VL%3DFISCHER%20Taschenbuch%20und%20AD%3D20140101%5E20150323	Search by publisher and last-updated date (from/to)
Batch search with date limit	5.6.2 Boolean searches Boolean searches and 5.6.3 Batch	POST	search=AD%3D20150319%5E20150320 { "content": [{ "isbn": "9783923*" },	A batch search by ISBN, domain and an additional change-format restriction (from/to)

Request type	Section	Type	Queries	Comments
	searches		{ "isbn": "405000*"}]	
Boolean keyword search	5.6.2 Boolean searches	GET	search=SW%3D GI%C3%BCck	Keyword search
Boolean keyword search with product-form restriction	5.6.2 Boolean searches	GET	search=ST%3DLinux%20 und%20PF%3Dnicht%20 E* Or: ST=Linux%20und%20PF %3Dnicht%20EA	Keyword search with a product-form restriction (not supported for digital products or e-books)
Boolean product-type search	5.6.2 Boolean searches	GET	search=VL%3DLangensc heidt%20und%20PT%3D not%20*Bundle or search=VL%3DLangensc heidt%20und%20PT%3D* Bundle	Select bundled products with VAT splitting, in conjunction with an additional Boolean parameter
EAN detail call	5.8 Product-entry retrieval	GET	product/4049817667903/ ean	EAN detail call
ISBN detail call	5.8 Product-entry retrieval	GET	product/9783775151061/ isbn13	ISBN detail call
Retrieving lists of product entries	5.7 Retrieving lists of product entries	POST	product/multipleProducts	
Index calls	5.11 Index searches	GET	index/author/meier	Accessing the author index for the "Meier" index entry
Accessing a cover in size "medium"	5.10 Cover retrieval	GET	cover/9783540537120/ m?	Cover authorization is needed in order to access cover data
Retrieving a cover in its original size (see section 5.10.2 Cover-Request)	5.10 Cover retrieval	GET	cover/9783729608887/ ?	Cover authorization is needed in order to access cover data
Requests for available media links to cover data and media objects	5.9 Accessing media files	GET	asset/mmo/ec94d0d04 71f4e6284f13f1a5f927c e3	A cover-permissions token is needed in order to access the list

6.3. API returns

Selected API cover or object calls for which authorizations apart from standard authorizations are required.

The list of error codes in the table is incomplete. Standard http error codes are used that are accompanied by a description of the error in question.

For further information on this topic, see section 5.4 Errors .

6.3.1. Login-function returns

API responses	"error": "unauthorized"	"error": "unauthorized"	"error": "unauthorized"
HTTP status	401	401	401
Error Description	No free slot in one of your registered stateful clients	Bad credentials	No stateful client registered for your credentials
	This account no longer has free API slots. You need to either close a token or wait for the timeout to end.	Erroneous login data.	The user is not authorized to use API. Contact MVB customer service.

6.3.2. Responses to metadata calls

The following contains a summary of (a) the messages that come into play for title metadata calls; and (b) a further overview of the messages that come into play for publisher metadata calls.

Interface messages may in some cases be determined by the access used:

- Statusless access
- Status-based access
- Access authorization for archived data

6.3.2.1. Responses to search calls

Example of call and API response	error": "invalid_token"	"error": "invalid_token"	"error": "unauthorized"	"error": "not_acceptable"	"error": "bad_request"
HTTP status	401	401	401	406	400
Error Description	Access token expired /	Invalid token does not	Full authentication	Could not find	Result window is too

Example of call and API response	error": "invalid_token"	"error": "invalid_token"	"error": "unauthorized"	"error": "not_acceptable"	"error": "bad_request"
HTTP status	401	401	401	406	400
	Invalid access token	contain resource id (metadata)	is required to access this resource	acceptable representation	large, page + size must be less than or equal to: [10000] but was [XXXXX]
api.vlb.de/api/v2/products?page=1&size=250&source=vlb&search=ST=Stichwort	Invalid access token*	Incorrect/invalid access token	The call was effected without a token.	Please enter <i>content type</i> and <i>accept</i> in the request header	the deep-paging protection has send this error message as the pagenumbers exceeds 10.000. for.ex.. page = 999999

*: The first call after a token expires returns *Access token expired*. The interface then outputs *Invalid access token*.

6.3.2.2. Responses to article data calls

API response/example of call	No data	error": "invalid_token"	"error": "invalid_token"	"error": "unauthorized"	"error": "forbidden"	"error": "not_found"	"error": "not_acceptable"	"error": "not_acceptable"
HTTP status	200	401	401	401	403	404	406	406
Error Description		Access token expired / Invalid access token	Invalid token does not contain resource id (metadata)	Full authentication is required to access this resource	Access is denied	product not found	Could not find acceptable representation	Could not find acceptable representation
api.vlb.de/api/v2/product/9783894410308/isbn13	Not allowed	Invalid access token**	Incorrect/invalid token	The call was effected without a token.	The token/access is not authorized to access this data ***	No ISBN/identifier or incorrect ISBN/identifier	Please enter <i>content type</i> and <i>accept</i> in the request header	For calls effected in ONYX format, the error message indicates that the article is not available in the desired format

**: The first call after the token expires returns *Access token expired*. The interface then outputs *Invalid access token*.

***: this error can occur if an account has no authorization to call archived products.

6.3.2.3. Publisher-data call returns

Example of call and API response	"error": "not_found"
HTTP status	404
Error Description	Cannot determine party with mvbId
api.vlb.de/api/v2/publisher/500553	No data available for this publisher ID

6.3.3. Returns for cover and media object calls

http status codes and API messages for cover and media object calls.

	Example of call and API response	No data	T101*	No_content	"error": "invalid_token"	"error": "unauthorized"	"error": "access_denied"	"error": "not_found"	"error"
HTTP status / call	HTTP status	200	200	204	401	401	403	404	503
Cover URL call	api.vlb.de/api/v1/cover/9783765732324/m?	Not allowed	Not allowed	Not allowed (cover call does not output 204)	Incorrect /invalid token	The account/token does not have archive permissions	The token/account is not authorized to access the cover	No ISBN/identifier or incorrect ISBN /identifier or no cover available	Temporarily unable to provide cover objects. Try again later.
Media-data link calls	api.vlb.de/api/v1/asset/mmo/ec94d0d0471f4e6284f13f1a5f927ce3	No media objects for this article	Not allowed	The product exists, but there are no media or cover objects for it.	Incorrect /invalid token	The account/token does not have media object permissions	The token/account is not authorized to call the list of media objects	Incorrect identifier	Temporarily unable to provide media objects. Try again later.

The endpoint of the cover and media request is the same as in the previous version of the interface (v1) – which means that media and cover data can be retrieved via either a v1 or a v2 URL.

6.4. Specimen data

Specimen data records in JSON-Format can be found via the [REST-API Beispieldaten](#) link, where they can be accessed as zip files.

The zip file contains examples of (a) outputs of detailed formats; and (b) results lists – in both cases as txt files.

7. Encoding

7.1. HTML encoding

All XML format values are encoded in HTML.

For further information on character encoding, see http://www.w3.org/MarkUp/html-spec/html-spec_13.html.

7.1. URL encoding

All URL parameters must be URL-encoded. For further information in this regard, see under RFC 1738 (<http://www.rfc-editor.org/rfc/rfc1738.txt>).

8. Contact information

General matters, access setup and technical queries concerning VLB REST API:

MVB customer service (MVB Kundenservice)

Phone: +49 69 1306-555

Fax +49 69 1306-255

kundenservice@mvb-online.de

MVB GmbH
Braubachstr. 16
60311 Frankfurt am Main

Postfach 10 04 42
60004 Frankfurt am Main

9. Annex

9.1. Standards and exceptions for sections and data fields

Advisory on results-lists and detailed-output fields:

- Kindly disregard any fields that are not found in these API specification. Such fields do not become valid until such time as they are incorporated into this document.
- In other words blank fields are no longer output, as “zero” values have been eliminated.
- Blank lists and blank fields are no longer output.
- Blank embedded objects can be output, provided that all of their fields are blank (e.g. “edition”: {},)
- The VLB aggregator aggregatorid is “zero” and the aggregatorld field is hidden. Thus the product aggregator is VLB.
- The *Contributor*-block date values are free text fields
- The *Identifier* field contains the main identifier for each product. In other words: for ÖSB titles, the Austrian school textbook number; for book titles, the ISBN, not hyphenated; and for non-book items the GTIN, not hyphenated.
- Repeatable sections are output in the order in which the values are to be displayed. This means that the sequence numbers are not output, except for a few sections. The sequences are based on the sequence of the output.
- Version 2 of REST API is based on ONIX 3.0 – and thus on the ONIX 3.0 code lists

9.2. Displays of products with VAT splitting (bundled products)

When products subject to VAT splitting are displayed, the following price processing requirements must be adhered to:

If the number of product parts (9.5.18) is the same as the number of tax elements within the price block in question (9.5.15), the productparts and taxes element groups can be correlated in their respective orders.

If the numbers are not identical owing to subsequent modification of the VAT splitting rule, the two element groups cannot be correlated with each other. This means that it is no longer possible to assign a tax rate to productparts. This scenario already occurs for non-DE prices – i.e. at present, USD and CHF prices, among others).

All other information in this regard will be provided if and when the VAT splitting rules are modified.

For information in this regard from the Börsenverein des Deutschen Buchhandels (German bookseller's association) see:

- [Mehrwertsteuer bei E-Bundle-Produkten \(value added tax information for E-bundle products\)](#)

9.3. Specimen data

Specimen data records in JSON-Format can be found via the [REST-API specimen](#) link (Beispiele V2), where they can be accessed as zip files.

The zip file contains examples of (a) outputs of detailed formats; and (b) results list – in both cases as txt files.

9.4. Note concerning results-lists fields

Following is a brief explanation of selected results-lists field:

Field name v2	Field type/value area/ ONIX code list	Mapping v1	Short / Long	Example	Comments
productId	Individual value	id	Short/Long	1d6a9158f1f24f67b572cf1b4b964463	Article UUID
version	Individual value	version	Long		

Field name v2	Field type/value area/ ONIX code list	Mapping v1	Short / Long	Example	Comments
identifier	Individual value	identifier	Long	4049817679876 or 978376573232 or SB170504	The <i>Identifier</i> field contains the main identifier for each product. Thus for ÖSB titles, the Austrian school textbook number; for book titles, the ISBN, not hyphenated; and for non-book items the GTIN, not hyphenated.
"isbn":	Individual value b244 / b221 (15)	"isbn":	Short/ Long	978-3-7657-3232-4	Hyphenated ISBN. Not for non-book products
issn	Individual value	issn	Short/Long		ISSN for journals (currently contains no data)
gtin	Individual value b244 / b221 (03)	gtin	Short/ Long	978376573232	Used for print titles, e-books, audio books and non-books
osbnr	Individual value	osbnr	Short/Long	SB170504	Austrian school-textbook number
Order no.	Individual value	Order no.	Short/Long	SB170504 or 1504-54 or MA19	publisher's order-number information
umbreitNummer	Individual value	umbreitNummer	Short/ Long	7512103	Book wholesaler's order no.
cbznumber	Individual appraisal date	cbznumber	Short/Long	18172985	Schweizer Buchzentrum (SBZ) order no.

Field name v2	Field type/value area/ ONIX code list	Mapping v1	Short / Long	Example	Comments
school textbook order no.	Individual appraisal date			1504-54 or MA19	order no. for school textbooks
title	Individual value b203 / b202 (01)	title	Short/ Long	Learn math: time	Article title
subTitle	Individual value B029	subTitle	Short/ Long	IFIP TC11 / WG11.3 Fifteenth Annual Working Conference on Database and Application Security July 15–18, 2001, Niagara on the Lake, Ontario, Canada	Article subtitle
publisher	Individual value	publisher	Short/ Long	Buch Verlag Kempen	Acronym for this publisher (i.e. Buch Verlag Kempen): VLB
publisherMvblId	Individual value	publisherMvblId	Short/ Long	89804 or 5234041	No publisher no.
publicationDate	b003		Short/ Long		in DD.MM.YYYY, MM.YYYY or YYYY format
edition	b058		Short/ Long	1st Corrected ed. 2006. Corr. 3rd printing 2006	
productAvailability	j396	availabilityStatePublisher	Short/ Long	20	Original availability from publisher indicated (j396)
productType		productType	Short/ Long	pbook	See section 9.4.1 productType values
priceEurD	Individual value	priceEurD	Short/ Long	6.95	Euro price, Germany
priceEurA	Individual value	priceEurA	Short/ Long	6.95	Euro price, Austria
priceCHF	Individual value	priceCHF	Short/ Long	12.5	CHF price, Switzerland
unpricedItemCode	j192			Null or 04	

Field name v2	Field type/value area/ ONIX code list	Mapping v1	Short / Long	Example	Comments
creationDate	Individual value	createDate	Short/ Long	24.03.2015	Product publication date
lastModificationDate	Individual value	lastModifiedDate	Short/ Long	24.03.2015	Last-updated date
active	boolean	state	Short/ Long	true	true or false
collections	Block	collections	Short/ Long		
collectionId		identifiers	Short/ Long	AAYBD53	Determine whether or not the item is a series
setId		identifiers		"X-3532-0783-7	Determine whether or not the item is a hierarchy
title	b203	title	Short/ Long	Book and bookseller statistics	Series or hierarchy title
subtitle	b029	subtitle	Short/ Long	Bookseller statistics	Series or hierarchy subtitle
partNumber		identifiers		3	Volume no.
productForm	List 150		Short/ Long		
productFormDetail	List 175		Short/ Long		
announcementDate		announcementDate	Short/ Long		
author		author	Short		
contributors	Block		Long		
type	List 17		Long		
firstName	Individual value		Long	Brigitte	First name
lastName	Individual value		Long	Neuböck-Hubinger	Last name
groupName	Individual value		Long		Company name

Field name v2	Field type/value area/ ONIX code list	Mapping v1	Short / Long	Example	Comments
priceAdditionEurD		priceAdditionEurD	Long	Suggested retail price	Identifier for recommended price
taxEurDInfo	Individual value		Long	0% VAT (information from publisher)	VAT (information from publisher)
taxKeyEurD	Individual value		Long	1	VAT key
priceCaEurD	Individual value		Long	Provisional	Provisional price in euros
priceProvisionalEurD	boolean		Short/ Long	true	True or false for provisional EurD price
priceFixedEurD	boolean		Short/ Long	true	True or false for fixed retail price
priceCalculatedEurD	boolean		Short/ Long		True or false
priceAdditionEurA	Individual value		Long	Suggested retail price	Identifier for recommended price
taxEurAInfo	Individual value		Long	0% VAT (information from publisher)	
taxKeyEurA	Individual value		Long	1	VAT key
priceCaEurA	Individual value		Long	Provisional	Provisional EurA price
priceProvisionalEurA	boolean		Short/ Long		True or false for provisional EurA price
priceAdditionChf	Individual value		Long	Suggested retail price	Identifier for recommended retail price
taxChfInfo	Individual value		Long	0% VAT (information from publisher)	
taxKeyChf	Individual value		Long	1	VAT key
priceCaChf	Individual value		Long	Provisional	Provisional price (in CHF)

Field name v2	Field type/value area/ ONIX code list	Mapping v1	Short / Long	Example	Comments
priceProvisionalChf			Long		True or false for provisional price (in CHF)
themaSubjects	Individual value	themaSubjects	Long	WTR	Topic content
themaQualifiers	Individual value	themaQualifiers	Long	1DFG	Topic supplement
keyWords	Individual value ONIX List 27 , value 20	keyWords	Long		Keywords
onSaleDate	Individual value	onSaleDate	Long		Date item went on sale
language	ONIX List 74	language	Long	eng / ger / fre	Article language
wholesaler		sigl	Long	C-BZ / A-MLO / U	Logogram/official seal information
aggregatorId		aggregatorId	Long	0 to 5001015	See section 5.13 Aggregators (overview)
coverUrl	Individual value	coverUrl	Long	http://api.vlb.de/api/v2/cover/9783142345678 /m	Link to a cover, if available
shortDescription	Individual value	shortDescription	Long		Short description limited to 1,500 characters
mainDescription	Individual value	mainDescription	Long		Main description limited to 1,500 characters
genreCode	Individual value	genreCode	Long	1295	Product group no.

9.4.1. *productType* values

The following values can potentially appear in the *productType* field

Product type	Product form b012 / explanation (partially in list 150)	Selectable via PT	ProductType (case sensitive)
Book	B*		pbook
e-book	E*		ebook
Audio/video	A*, V*		abook
Calendar	PC		calendar
cartographic material	C*		map
Digital product	D*, but not E*		Digital
Non-Book	All other values apart from 00		nonbook
Undefined	00 or undefined		undefined
Series			Series
Hierarchy			Hierarchy
Duo bundle	No ONIX equivalent	Yes	duoBundle
multi-component bundle	No ONIX equivalent	Yes	multiBundle
Undefined	No ONIX equivalent		Undefined

9.4.2. Genre codes: advisory on product-group classification

Genre codes in results lists are product groups for specific products.

The same product group classification is also used in the *subjects* detail block for type 26.

An overview of the product group system can be found at www.vlb.de – under *Verlage – Downloadcenter*. The relevant links are as follows:

PDF with explanations: https://vzb.de/assets/images/wgsneuversion2_0.pdf

or (Excel file, codes only, plus translations): https://vzb.de/assets/images/wgsneu_mit_index_20060802.xlsx General information on product classification: <https://vzb.de/hilfe/vzb-verlag/titelerfassung/produktklassifikation>

9.4.3. Reference price designations

The dedicated reference-price designation field was dropped as of version 2 of REST API. The designations can be found by combining the following fields

Price field	EUR-D	EUR-A
priceProvisionalEurD	false	
priceFixedEurD	true	
priceProvisionalEurA		false
priceFixedEurA		true

A price can be considered a reference price only insofar as no provisional price designation is available for it and the price is a fixed retail price.

9.5. Overview of the requisite ONIX code lists

The tables below list the fields used in VLB REST API V2.

These tables are mainly intended for v2 users. In the interest of easing the transition from v1 to v2, the v1 fields have also been included.

9.5.1. Unclassified fields

The following fields can be found at the various product levels and are unclassified.

Element	V2	O30 list	Repeatable	O30-short	V1	O21 list	O21-short
Article status	active				active		
Individual productID	productId			a001	id		a001
ID from the old system	bvdId				bvdId		
Product aggregator	aggregatorId				aggregatorId		
Date of first publication	publicationDate			x448 = 01 b306	publicationDate		b003
Announcement date	announcementDate			x448 = 09 / B306	announcementDa		B086

Element	V2	O30 list	Repeatable	O30-short	V1	O21 list	O21-short
					<i>te</i>		
Availability date Availability date	availabilityDate			x461 = 08 / B306	<i>availabilityDate</i>		<i>j142</i>
On-sale date	onSaleDate			x448 = 02 B306	<i>onSaleDate</i>		<i>j143</i>
Year of publication	yearFirstPublished			x448 = 20b306			
Thesis type	thesisType			b368	<i>thesisType</i>		<i>b368</i>
Institution where the thesis was presented	thesisPresentedTo			b369	<i>thesisPresentedTo</i>		<i>b369</i>
Thesis year	thesisYear			b370	<i>thesisYear</i>		<i>b370</i>
Country of manufacture	countryOfManufacture				<i>countryOfManufacture</i>		
Product availability	productAvailability	List 65		J396	<i>availability</i>		<i>J396</i>
Publication status	publishingStatus	List 64		b394	<i>publishingStatus</i>		
Publication status	publishingStatusNote			b395	<i>publishingStatusNote</i>		
Product type	productType				<i>productType</i>	-	-
ONIX bundle flag	productComposition	List 2		X314	-	-	-
Unpriced item	unpricedItemCode				<i>unpricedItemCode</i>		
Article with unspecified author	noContributor				<i>noContributor</i>		
Product status in a specified market. Needed for promotionCampaign	marketPublishingStatus	List 68		J407			
Free text for description of product's PR campaign	promotionCampaign			k165			
City of publication	publicationCity		Yes	B209	<i>publicationCity</i>		
Country of publication	publicationCountry			B083	<i>publicationCountry</i>		
Special-edition designation	tradeCategory	List 12		B384	<i>tradeCategory</i>	List 12	<i>B384</i>
Date created	creationDate				<i>createdDate</i>		
Last-updated date	lastModificationDate				<i>lastModifiedDate</i>		
Sequentially generated	bitNo				<i>bitNo</i>		

Element	V2	O30 list	Repeatable	O30-short	V1	O21 list	O21-short
number						-	-
Indication as to whether the article is subject to VAT splitting.	splitVat				bundle	-	-
Minimum order	orderQuantityMinimum			x532			
Packing unit	orderQuantityMultiple			x533			

9.5.2. *taxKeyEurD / taxKeyEurA / taxKeyChf*

The following values can potentially appear in the *taxKeyXX* field.

taxkey	ONIX code	Meaning
0	Z	No VAT information
1	R	Reduced VAT rate
2	S	Full VAT rate
6		VAT bundle designation

9.5.3. *ancillarycontents – illustrations*

Detailed information is available via VLB help, under [Illustrations](#).

The content of this block is repeatable.

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Type of illustration	ancillaryContentType	List 25	x423	type	List 25	b256
Description of illustration	ancillaryContentDescription		x424	description		b361
Number of illustrations	number		b257	number		b257

9.5.4. audiences and recommended age group

Detailed information is available via VLB help, under [audience and recommended age group](#).

The content of this block is repeatable.

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Description of target audience	audienceDescription		b207	description		b207
Type of age group recommendation	audienceRangeQualifier	List 30	b074	rangeCode	List 30	b074
from-to information	audienceRangeAgeFrom	List 31	b075	ageFrom	List 31	b075
from-to value	audienceRangeAgeTo		b076	ageTo		b076
Recommendation system	audienceCodeType	List 29	b204	type	List 29	b204
Name of recommendation system	audienceCodeTypeName		b205	codetypename		b205
Value in recommendation system	audienceCodeValue		b206	value		b206

9.5.5. citedContents

Detailed information is available via VLB help, under [cited contents](#).

The content of this block is repeatable.

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Ancillary text-type code	citedContentType		x430	-		

Element	v2	O30 list	O30-short	v1	O21 list	O21-short
Ancillary-text audience	citedContentAudience		x427	-		
Name of bestseller list	listName		x432	-		
Position on bestseller list	listPosition		x433	-		
Link 1 to text	resourceLink1		x435	-		
Link 2 to text	resourceLink2		x435	-		
Link 3 to text	resourceLink3		x435	-		
Source title	sourceTitle		x428	-		
Source type	sourceType	List 157	x431	-		
Comments	citationNote		x434	-		
First edition	publicationDate		x429=01; b306	-		
Valid as from	validFrom		x429=14/24; b306	-		
Valid until	validUntil		x429=15/24; b306	-		
Last updated	lastUpdated		x429=17; b306	-		

9.5.6. Series, series and sets

Detailed information is available via VLB help, under [series and hierarchies](#).

The content of this block is repeatable.

Element	v2	O30 list	O30-short	v1	O21 list	O21-short
Series ID	collectionId		b244	Type		b244
Hierarchy ID	setId			Active		b244 b233=VLB-Xnr
Series or hierarchy title	title		b203 with x409=02	Title		b203 or b018 b203 or b023

Element	v2	O30 list	O30-short	v1	O21 list	O21-short
Subtitle	subtitle		b029 with x409=02	Subtitle		b029
Volume no.	partNumber		x410 with x409=01	sequence		

9.5.7. contributors - Author

Detailed information is available via VLB help, under [author](#).

The content of this block is repeatable.

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Author enumeration	sequenceNumber	-	b034	sequence		b034
Author type	contributorRole	List 17	b035	role	List 17	b035
Translated from	fromLanguage	List 74	x412	language	List 74	b252
Translated into	toLanguage	List 74	x413			-
Name type	nameType	List 18	x414			-
GND		List 44	nameidentifier/		List 101	personnameide ntifier/
	gnd	-	x415=25	dnbid		b390=25
		-	b244			b244
ISNI	isni	List 44	nameidentifier/		List 101	personnameide ntifier/
		-	x415=16			b390=16
		-	b244			b244
ORCID		List 44	nameidentifier/			-
	orcid		x415=21			
		-	b244			
First name	firstName	-	b039	firstName		b039

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Last name	lastName	-	b040	lastName		b040
Date of birth		List 177	contributordate/		List 75	persondate/
	dateOfBirth	-	x417=50	dateOfBirth		b305=007
	dateOfBirthFormat	-	b306 with dateformat	dateOfBirthFormat		j260
		-				b306
Date of death		List 177	contributordate/		List 75	persondate/
	dateOfDeath	-	x417=51	dateOfDeath		b305=008
	dateOfDeathFormat	-	b306 with dateformat	dateOfDeathFormat		j260
		-				b306
Profession		-	professionalaffiliation/			professionalaffiliation/
	professionalPosition	-	b045	professionalPosition		b045
Company name		-	professionalaffiliation/			professionalaffiliation/
	professionalAffiliation	-	b046	professionalPosition		b046
Corporation name	corporateName	-	b047	groupName		b047
Biographical information	biographicalNote	-	b044	biographicalNote		b044

A publisher node can also contain subnodes 9.5.27 websites in order to specify a publisher URL.

9.5.8. *copyrights*

Detailed information is available via VLB help, under [copyright](#).

The content of this block is repeatable.

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Copyright type	copyrightType	List 219	x512	-		
Copyright year	copyrightYear	-	b087	-		
Name of first copyright holder (person)	personName1	-	b036	-		
Name of second copyright holder (person)	personName2	-	b036	-		
Name of third copyright holder (person)	personName3	-	b036	-		
Name of first copyright holder (corporation)	corporateName1	-	b047	-		
Name of second copyright holder (corporation)	corporateName2	-	b047	-		
Name of third copyright holder (corporation)	corporateName3	-	b047			

9.5.9. Editions

Detailed information is available via VLB help, under [editions, city of publication and country of publication](#).

The *edition* block does not contain a complete listing of all ONIX fields.

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Type of edition	editionType	List 21	x419	type	List 21	b056
Edition no.	editionNumber	-	b057	number		b057
Edition text	editionStatement	-	b058	text		b058
Edition version	editionVersionNumber	-	b217	release		B217

9.5.10. *extent - Extent*

An overview of the characteristics of the product that are bundled into the extent-related information can be found in the *extent* block.

An overview of all of the fields listed below can be found in the *extent* block. In the interest of clarity, the various tables are provided individually.

Concerning lists 23 and 24 below, please note that list 23 pertains to type and list 24 to unit.

9.5.10.1. *Page counts*

Detailed information is available via VLB help, under [page counts](#).

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Total page count	contentPageCount	List 23 / List 24	b219 with b218=11 and b220=03	pages		b061
Main-content page count	mainContentPageCount	List 23 / List 24	b219 with b218=00 and b220=03	pagesArabic		b255
Front-matter page count	mainContentPageCount	List 23 / List 24	b219 with b218=00 and b220=03	pagesRoman	List 23	b218=03
Back-matter page count	backMatterPageCount	List 23 / List 24	x421 with b218=04 and b220=03			
Numbered-pages count	totalNumberedPages	List 23 / List 24	b219 with b218=05 and b220=03			
Production page count	productionPageCount	List 23 / List 24	b219 with b218=06 and b220=03			
Unnumbered-page count	unnumberedInsertPageCount	List 23 / List 24	b219 with b218=12 and b220=03			
Exact page count	absolutePageCount	List 23 / List 24	b219 with b218=07 and b220=03			
Related print-product page count	printCounterpartPageCount	List 23 / List 24	b219 with b218=08 and b220=03			
Provisional page count	digitalProductPageCount	List 23 / List 24	b219 with b218=10 and b220=03			

9.5.10.1. DRM type

Detailed information is available via VLB help, under [DRM type](#).

Element	V2	O30 list	O30-short	V1	O21-list	O21-short	O21 list
DRM type	drmCode	List 144	x317	drmCode			List 144

9.5.10.2. File size

Detailed information is available via VLB help, under [file size](#).

Element	V2	O30 list	O30-short	V1	O21-list	O21-short
File size	fileSize	List 23 / List 24	b219 with b218=22 and b220=17/18/19	fileSize	List 23	b218=22

File size (filesize) is normalized to bytes ([List 24](#)).

9.5.10.3. Duration, track count, map scale

Detailed information is available via VLB help, under [duration, track count, map scale](#).

Element	V2	O30 list	O30-short	V1	O21-list	O21-short
Duration	duration	List 23 / List 24	b219 with b218=09 and b220=04/05/06/14/15/16	duration	List 23	b218=09
Track count	numberOfTracks	List 23 / List 24	b219 with b218=09 and b220=11	numTracks	List 23	b218=09
Map scale	mapScale	-	b063	mapScale		b063

Duration is normalized to minutes ([List 24](#) Code 05).

9.5.10.4. *Illustrations*

Detailed information is available via VLB help, under [Illustrations](#).

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Total number of illustrations	numberOfIllustrations	-	b125	numillustrations		b125
Captions	illustrationNote	-	b062	illustrationText		b062

9.5.11. *Product form*

Detailed information is available via VLB help, under [Product form](#).

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Product form	productForm	List 150	b012		List 7	b012
Product form details	productFormDetail	List 175	b333	binding	List 78	b333
Main product content	primaryContentType	List 81	x416			-
Product content	productContentType	List 81	b385	productContentT ype	List 81	b385
VLB single-volume type	productFormDescription	-	b014	description		b014
Packaging	productPackaging	List 80	b225	packaging	List 80	b225
Height	height	List 48	x315=01	height	List 48	c093=01
		List 50	c095=mm		List 50	c094
		-	c094			c095=mm
Width	width	List 48	x315=02	width	List 48	c093=02
		List 50	c095=mm		List 50	c095=mm

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
		c094			c094	
Thickness	thickness	List 48	x315=03	thickness	List 48	c093=03
		List 50	c095=mm		List 50	c095=mm
		-	c094			c094
Diameter	diameter	List 48	x315=09/12	diameter	List 48	c093=09/12
		List 50	c095=mm		List 50	c095=mm
		-	c094			c094
Weight	weight	List 48	x315=08	weight	List 48	c093=08
		List 50	c095=g		List 50	c094
		-	c094			c095=g

9.5.12. *formFeatures warning*

Detailed information is available via VLB help, under [Warning](#).

The content of this block is repeatable.

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Warning type	productFormFeatureType	List 184	b334=13	type	List 184	b334=13
	productFormFeatureValue	-	b335	code		b335
Warning		-	b334=13			b334=13
	productFormFeatureDescription	-	b336	description		b336

9.5.13. *Product number identifiers*

Detailed information is available via VLB help, under [Product numbers](#).

The content of this block is repeatable.

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
ID type	productIdentifierType	List 5	b221	type	List 5	b221
ID name	idTypeName	-	b233	typeName		b233
Value	idValue	-	b244	value		b244

9.5.14. *Languages/product languages*

Detailed information is available via VLB help, under [title and product language](#).

The content of this block is repeatable.

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Language type	languageRole	List 22	b253	type	List 22	b253
Language	languageCode	List 74	b252	value	List 74	b252

9.5.15. *Prices*

Detailed information is available via VLB help, under [Prices](#).

The content of this block is repeatable.

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Price type	priceType	List 58	x462	type	List 58	j148
Price qualifier	priceQualifier	List 59	j261	typeQualifier	List 59	j261
Price notes	priceTypeDescription	-	j262	typeDescription		j262
Provisional price	priceStatus	List 61	j266	state	List 61	j266
Total price	priceAmount	-	j151	value		j151
Indicates whether the price has been calculated	calculated	-				
section 9.5.15.1.1	taxes	-				
Currency	currencyCode	List 96	j152	currency	List 96	j152
Country	countriesIncluded	List 91	territory/ x449	country	List 91	b251
Geographical entity	regionsIncluded	List 49	territory/ x450	territory	List 49	j303
Valid as from	validFrom	List 173	PriceDate/ x476=14/24 b306	validFrom		j161
Expiration date	validUntil	List 173	PriceDate/ x476=15/24 b306	validUntil		j162

9.5.15.1.1. Taxes (*included in the price block*)

The *taxes* block is available solely in the price block.

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
VAT rate code	taxRateCode	-	x471		List 62	
VAT rate	taxRatePercent	-	x472			
Taxable price	taxableAmount	-	x473			
Tax amount	taxAmount	-	x474			

9.5.16. Prices

Detailed information is available via VLB help, under [Prices](#).

The content of this block is repeatable.

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Price name	prizeName	-	g126	-		
Price year	prizeYear	-	g127	-		
Price country	prizeCountry	List 91 (ISO 3166-1 2)	g128	-		
Price type	prizeCode	List 41	g129	-		
Comments	prizeStatement	-	x503	-		
Jury/organization	prizeJury	-	g343	-		

9.5.17. *productClassifications – customs code number*

Detailed information is available via VLB help, under [Customs tariff code number and country of manufacture](#).

The content of this block is repeatable.

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Produktklassifikationscode	productClassificationType	List 9	b274=04	type		b274=04
Productklassifikation	productClassificationCode	-	b275	value		b275

Most tariff code numbers can be found here.

9.5.18. *productParts – (v1: containeditems) – product elements and ancillary elements*

Detailed information is available via VLB help, under [Product elements and ancillary elements](#).

The content of this block is repeatable.

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Main element	primaryPart		x457			-
Product-number type:	productIdentifierType	List 5	productidentifier/ b221	productIdentifierType	List 5	productidentifier/ b221
Type description for product number	productIdentifierTypeName	-	b233	productIdentifierTypeName		B233
Product number	productIdentifierValue	-	productidentifier/ b244	productIdentifierValue		productidentifier/ b244
Product form	productForm	List 150	b012	type	List 7	b012
Product form details	productFormDetail	List 175	b333		List 78	b333
Description	productFormDescription	-	b014			b014
Product content	productContentType	List 81	b385	productContentType	List 81	b385
number of parts	numberOfItemsOfThisForm	-	x322	numberOfPieces		b210
Number of copies	numberOfCopies	-	x323			b015

9.5.19. *publisherData*

The *publisherData* block is only made available to statusless tokens after being activated by MVB Customer Service.

JSON field	Example	Comments
mvbld	5106488	Publisher's MVB ID (same as request parameter)
shortName	MVB	Publisher's short name
name	MVB Marketing- und Verlagsservice des Buchhandels GmbH	Publisher's full name
street	Braubachstr. 16	Street address
cityStreet		City/town
zipStreet		Postal code
country	D	Country
postbox	10 04 42	Publisher's PO box
phone	+49 (069) 1306-550	Phone
fax:	+49 (069) 1306-255	Fax
email	vertrieb@mvb-online.de	Mailing address
url	www.mvb-online.de	URL of publisher's website
isbnPrefixes	978-3-7657,978-3-65	All ISBN prefixes that are assigned to this publisher (as a block)
deliveryBS	Zero	Wholesale bookseller's delivery information
deliveryNote	„N“	SOA delivery advisory (- E: "Publisher sells all or part of its output to end customers only." - R: "In exceptional cases, publisher sells all or part of its output to wholesalers, without a discount." - N: "Publisher sells all or part of its output solely on a pro forma or COD basis."
bagCreditor	False	Invoicing via the BAG (true/false)
bagDebtor	False	Invoicing via the BAG (true/false)
tax	DE114130036	Publisher's VAT ID no.
siglDe		German logogram/official seal
siglAt		Austrian logogram/official seal
siglCh		Swiss logogram/official seal
vnrKreditor	10871	Creditor transaction no.
vnrKreditorAt		Creditor transaction no. in Austria
vnrKreditorCh		Creditor transaction no. in Switzerland
gln	4018544000000	

9.5.20. *Publishers*

Detailed information is available via VLB help, under [Publishers and imprints](#).

The content of this block is repeatable.

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Publishing role	publishingRole	List 45	b291	type	List 45	b291
Publisher's name	publisherName	-	b081 or b079	name		b081 or b079
Name in German booksellers' association address book	adbName	-		adbName		
Identifier type	publisherIDType	List 44	publisheridentifier/ x447	nameCodeType	List 44	b241
Identifier	iDValue	-	publisheridentifier b244 b295	publisherid		b243 b295

A publisher node can also contain subnodes 9.5.27 websites in order to specify a publisher URL

9.5.21. *Related products*

Detailed information is available via VLB help, under [Related products](#).

The content of this block is repeatable.

Element	V2	O30 list	O30-short	V1	O21 list	O21-short

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Product relation type	productRelationCode	List 51	x455	Type	List 51	h208
Product-number type	productIdType	List 5	productidentifier/ b221	productIdType	List 5	productidentifier/ b221
Product number	productIdValue	-	productidentifier/ b244	productIdValue		productidentifier/ b244
Product form	productForm	List 150	b012	productForm	List 7	b012
Product form details	productFormDetail	List 175	b333	productFormDetail	List 78	b333

9.5.22. *SalesRight*

Detailed information is available via VLB help, under [SalesRights](#).

The content of this block is repeatable.

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Type of sales (i.e. marketing) rights	salesRightsType	List 46	B089	type	List 46	b089
Country	countriesIncluded	List 91	territory/ x449	country	List 91	b090
Geographical entity	regionsIncluded	List 49	territory/ x450	territory	List 49	b388

9.5.23. *subjects – product classifications*

Detailed information is available via VLB help, under [Product classification](#).

The content of this block is repeatable.

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Main product classification	mainSubject	-	x425	main		-
Classification-source name	sourceName	-		sourceName		
Product-classification type	subjectSchemeIdentifier	List 27	b067	type	<i>List 26 or List 27</i>	<i>b191 or b067.</i>
Product-classification version no.	subjectSchemeVersion	-	b068	systematicVersion		b068
Product classification name	subjectSchemeName	-	b171	schemeName		b171
Code	subjectCode	-	b069	code		b069
Name	subjectHeadingText	-	b070	value		b070

9.5.24. *supportingResource media files*

Detailed information is available via VLB help, under [Media files](#).

The content of this block is repeatable.

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Media file type	resourceContentType	List 158	x436	type	List 38	mediafile/ f114
Target group	contentAudience	List 154	x427			
Media type	resourceMode	List 159	x437			-
Resource form (link, download etc.)	resourceForm	List 161	x441			
Credits	credit	List 160	x438=01; x440			
Subtitle	caption	List 160	x438=02; x440			

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Copyright	copyright	List 160	x438=03; x440			
Copyright holder	copyrightHolder	List 160	x438=03; x439			
Contributor's ISNI	contributorIsni	List 160	x438=05; x440			
Proprietary ID of the relevant contributor – where possible, as per sequence no.	contributorSequenceNumber	List 161	x438=06; x441			
Duration	duration	List 162	x438=04; x439			
File format	fileFormat	List 178	x442=01; x439			
Image height	imageHeight	List 162	x442=02; x439			
Image width	imageWidth	List 162	x442=03; x439			
File name	filename	List 162	x442=04; x439			
File size in megabytes	filesizeApprox	List 162	x442=05; x439			
File size in bytes	filesizeExact	List 162	x442=07; x439			
MD5 hash value	md5Hash	List 162	x442=06; x439			
SHA256 hash value	sha256Hash	List 162	x442=08; x439			
First edition	publicationDate	List 155	x429=01; b306		d109	
Valid as from	validFrom	List 155	x429=14/24; b306		b324	
Valid until	validUntil	List 155	x429=15/24; b306		b325	
Last updated	lastUpdated	List 155	x429=17; b306		f373	
External link 1	exportedLink	-	x435			mediafile/ f117 or othertext/d106

Reading rationales (Lesemotive) (B067 = B8)

CodeNr	Reading rational classification text
01	Auseinandersetzen
02	Eintauchen
03	Entdecken
04	Entspannen
05	Lachen

CodeNr	Reading rational classification text
06	Leichtlesen
07	Nervenkitzeln
08	Optimieren
09	Orientieren
10	Verstehen

To use the reading rationales the german classification text as shown in the table have to be used.

Akutelle Liste mit Erläuterungen unter: <https://vlb.de/hilfe/lesemotive>

9.5.25. *textContents – ancillary texts*

Detailed information is available via VLB help, under [Ancillary texts](#).

The content of this block is repeatable.

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Type of ancillary text	textType	List 153	x426	type	List 33	d102
Target audience	textContentAudience	List 154	x427			-
Textformat	textFormat	List 34	d104 – textformat attribute	formatcode	List 34	d103
Language	language	List 74 (ISO 639-2/B)	d104 with language	language		d104 with language
Text	text	-	d104			
Author 1	textAuthor1	-	d107			-
Author 2	textAuthor2	-	d107			-

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Author 3	textAuthor3	-	d107			-
Organization	textSourceCorporate	-	b374			-
Title	sourceTitle	-	x428			-
publication date	PublicationDate	List 155	contentdate/ x429=01 b306	textPublication Date		d109
Valid as from	validFrom	List 155	contentdate/ x429=14/24 b306	validFrom		b324
expiration date	validUntil	List 155	contentdate/ x429=15/24 b306	validUntil		b325
Last updated	lastUpdated	List 155	contentdate/ x429=17 b306			-

9.5.26. *Titles and product languages*

Detailed information is available via VLB help, under [titles and title languages](#).

Unlike ONIX 3.0, the product language in REST API can be retrieved in block 9.5.14 Languages/product languages.

The content of this block is repeatable.

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Type of title	titleType	List 15	b202	type	List 15	b202
Title	title	-	titleelement/ x409=01 b203	title		b203
Subtitle	subtitle	-	titleelement/ x409=01 b029	subtitle		b029

9.5.27. *websites*

Detailed information is available via VLB help, under [Product websites](#).

The content of this block is repeatable.

In ONIX 3.9 and REST API v2, websites are available solely as sub-sections.

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Website type	websiteRole	List 73	b367	type	List 73	b367
Website text	websiteDescription	-	b294	description		b294
Link to website	websiteLink	-	b295	url		b295

9.5.28. *Wholesalers*

The content of this block is repeatable.

Element	V2	O30 list	O30-short	V1	O21 list	O21-short
Order number	orderNumber	-	-	orderNumber		
Logogram/official seal designation	sigl	-	-	sigl		

9.5.29. *productContacts – contact address and product safety*

Detailed information are available through the VLB-help pages in chapter [Kontaktadressen und Produktsicherheit](#) (contact address and product safety – the webpage is currently only in german).

The content of this blockes is repeatable.

Element	V2	O3-Liste	O3-Short	V1	O21-Liste	O21-Short
Type of contact	type	List 198	x482			
organisation	organization	-	x484			
person	person	-	x299			
phone	phone	-	j270			
e-mail	mail	-	j272			
street, street number.	streetAddress	-	x552			
city	location	-	j349			
Postal code	postalCode	-	x590			
country	country	List 91 (ISO-3166-1)	b251			